The Implementation of a 2-Core, Multi-Threaded Itanium Family Processor

Samuel Naffziger, *Member, IEEE*, Blaine Stackhouse, *Member, IEEE*, Tom Grutkowski, Doug Josephson, Jayen Desai, Elad Alon, *Senior Member, IEEE*, and Mark Horowitz, *Fellow, IEEE*

***Abstract*—The design of the high end server processor code named Montecito incorporated several ambitious goals requiring innovation. The most obvious being the incorporation of two legacy cores on-die and at the same time reducing power by 23%. This is an effective 325% increase in MIPS per watt which necessitated a holistic focus on power reduction and management. The next challenge in the implementation was to ensure robust and high frequency circuit operation in the 90-nm process generation which brings with it higher leakage and greater variability. Achieving this goal required new methodologies for design, a greatly improved and tunable clock system and a better understanding of our power grid behavior all of which required new circuits and capabilities. The final aspect of circuit design improvement involved the I/O design for our legacy multi-drop system bus. To properly feed the two high frequency cores with memory bandwidth we needed to ensure frequency headroom in the operation of the bus. This was achieved through several innovations in controllability and tuning of the I/O buffers which are discussed as well.**

I. Introduction

THE next generation in the Itanium processor family, code named Montecito, has two dual-threaded cores integrated on die with 26.5 MB of cache in a 90-nm process with seven layers of copper interconnect. The die is 21.5 mm by 27.7 mm and includes 1.72 billion transistors. With both cores at full frequency, it consumes 100 W total, allocated to four separate power planes; $V_{\rm core}$ for the two cores, $V_{\rm cache}$ for the 24 MB data arrays, $V_{\rm fixed}$ for bus interface logic, and $V_{\rm tt}$ for I/O driver termination. The micro-architecture and circuit methodologies are directly leveraged from the prior Itanium2 processors [1] but contain significant improvements for robustness in the 90-nm process. Architectural additions to the design include the integration of two cores on-die, each with a dedicated 12 MB third level cache, a 1 MB second level instruction cache and dual threading (Fig. 1). Susceptibility to soft errors is also reduced with design changes and power efficiency is also improved through low power design techniques and active power management [2].

II. Silicon Power Measurements

Montecito power measurements were collected on both system and tester platforms. Fig. 2 depicts system power measurements for a typical part running in fixed frequency mode (FFM) at 1.2 V and 2.0 GHz. In addition to the SpecInt and SpecFP application suites, worst case (aka virus) power is also plotted in this figure. The code for this pathological case was constructed with no other purpose but to burn power. On this particular part, running the virus case or the peak portions of the FP Spec will dissipate greater than the 85 W allocated to the $V_{\rm core}$ dissipation. In order to pull these high power applications in line with the power specification, a traditional microprocessor design would be forced to drop its frequency across the entire application space.

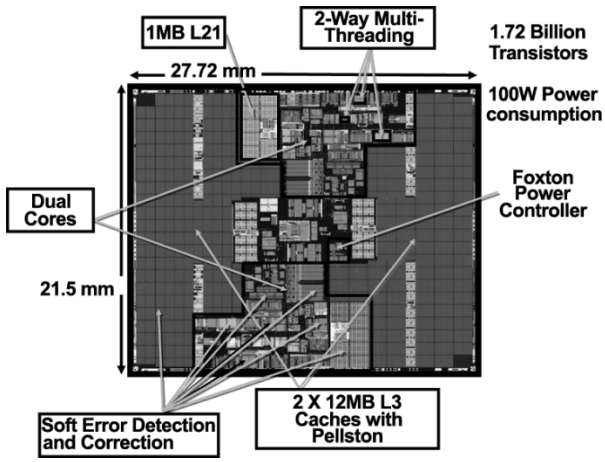


Fig. 1. Processor overview.

A. Application Power Variability

From Fig. 2, it is evident that between many applications, the average power dissipation level varies over a large range. Fig. 3 shows a time-resolved data collection for one of the SpecInt applications, int.256.bzip2. The application was run under Linux and the resolution of the data collection is 50 ms; this shows that within applications, the power consumption varies widely as well.

Data collection across a wide spread of material gives better insight into the power dissipation issue. Fig. 4 shows a scatter plot of floating-point power dissipation (peak of fp.178.GALGEL application power) across a large sample of units. Power dissipation data for both 1.8 and 2.0 GHz operation is depicted. The horizontal axis of this graph is the average frequency (in megaherts) of a 19-stage on-die ring oscillator used to benchmark the performance of the underlying silicon. All of the material will consume greater then 85 W

Manuscript received May 16, 2005; revised August 22, 2005.

S. Naffziger, B. Stackhouse, T. Grutkowski, D. Josephson, and J. Desai are with Intel Corporation, Fort Collins, CO 80528 USA (e-mail: sam.naffziger@intel.com).

E. Alon and M. Horowitz are with Stanford University, Stanford, CA 94305 USA.

Digital Object Identifier 10.1109/JSSC.2005.859894

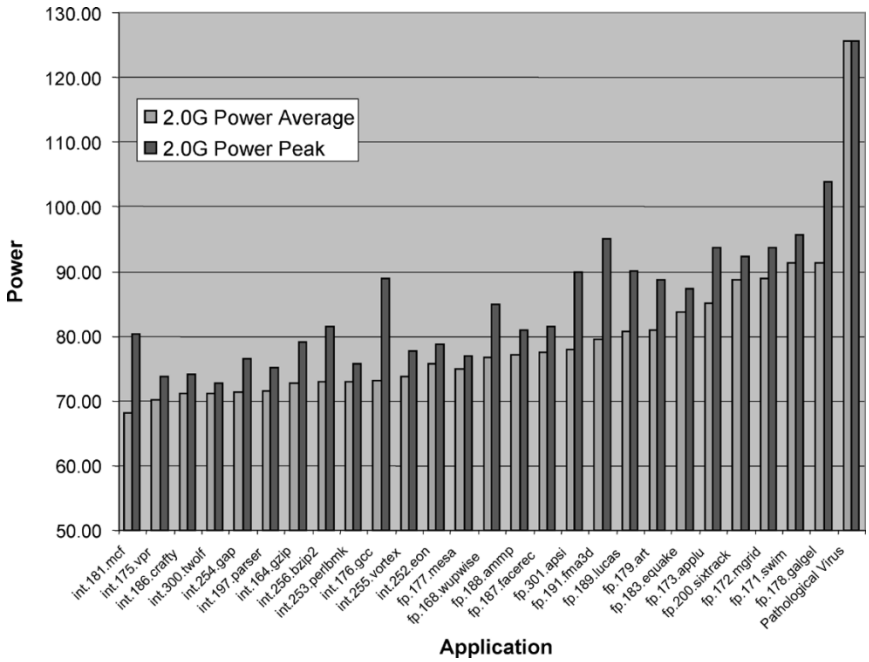


Fig. 2. Power measurement across benchmark suite.

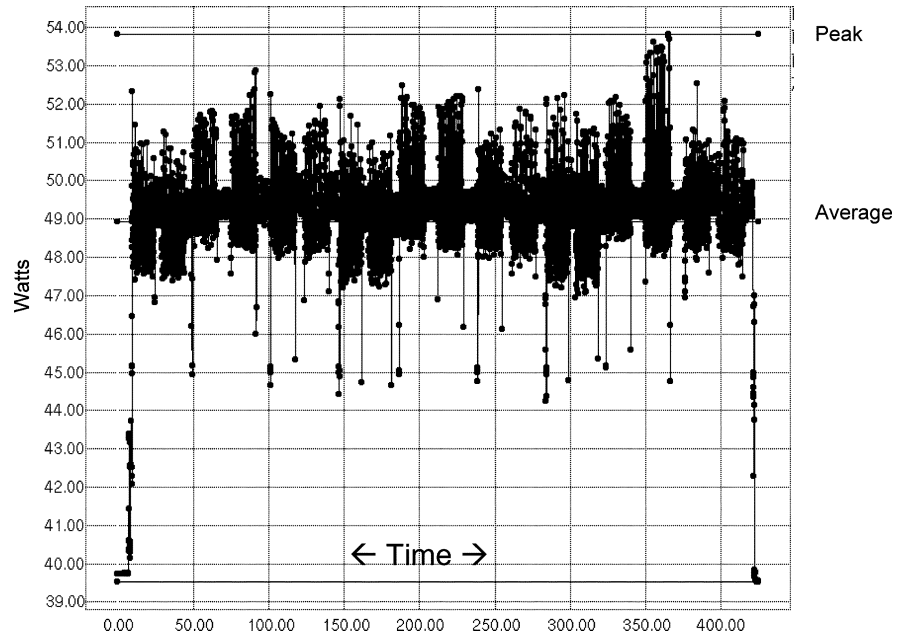


Fig. 3. Time resolved relative current draw for int.256.bzip2.

when running the application at 2.0 GHz and 1.2 V. Even at the reduced 1.8-GHz frequency, a majority of the parts will dissipate greater than 85 W with a 1.2-V core supply voltage. Hence the motivation for the power measurement and dynamic management capabilities present in the Foxton design. As discussed above, Foxton will move to reduce supply voltage in order to bring high-end floating point applications under the 85-W limit. Fig. 5 depicts the voltage to which the Foxton controller drives the part, forcing the application back under the 85 W $V_{\rm core}$ power specification. In addition to the $V_{\rm core}$ voltage for GALGEL running at 1.8 Ghz, the average voltage for EON running at 2.0 GHz, and the power virus running at 1.8 GHz is also plotted. Part for part, despite the higher frequency, EON at 2.0 GHz is able to stay within the 85 W at a measurably higher voltage than GALGEL. It should be noted that process Vmax limitations will limit the high-end voltage to which Foxton will be allowed to drive the part. This is particularly true for the slowest material in the distribution The typical frequency delta from EON to GALGEL is 6%–7% which represents the greatest frequency reduction for realistic applications. For the worst case power-virus code, the reduction is $\sim$15%. Even with reliability-induced Vmax limitations, the voltage data illustrated in Fig. 5 is sufficient to support a sizable frequency boost for most midrange power applications.

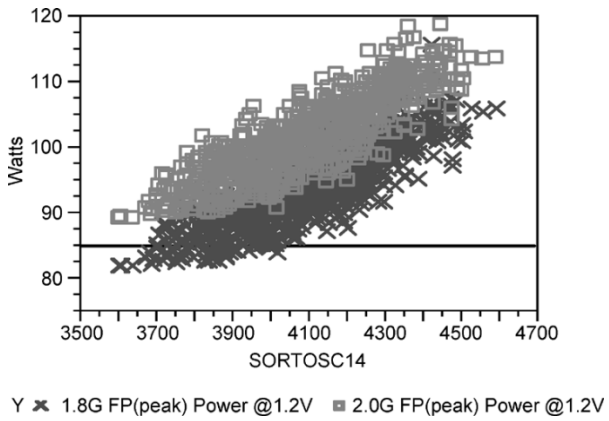


Fig. 4. Galgel peak power at 1.2 V.

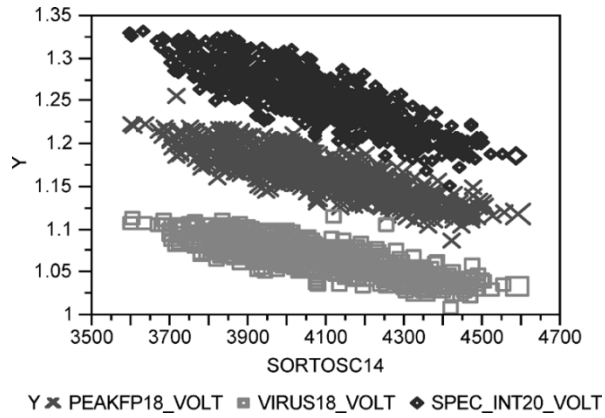


Fig. 5. Application voltage.

B. Application Frequency and Binning Concerns

A Montecito unit that is binned at 2.0 GHz (Fceiling $=$ 2 GHz) will operate at this maximum frequency during most low and mid-power applications. It is also required that parts meet or exceed a *base* frequency across the entire range of useful applications. In addition, part to part variation of this base frequency must be minimized to provide consistent performance across parts and operating conditions. The first step taken to minimize variation is to *de-rate* the power envelope for those parts capable of hitting 2.0 GHz frequency. Any part operating below 85 W for the $bin - application$ has its power reference $(Pref)$ reduced to the minimum power level required to support the binning application at 2.0 GHz. Fig. 6 shows the de-rated $Pref$ for a distribution of parts binned to 2.0 GHz (bin application $=$ EON). This procedure has the effect of reducing part to part frequency variation for higher power applications. Fig. 7 depicts the base frequency for the same distribution of material. The manufacturing process will make one additional optimization, adjusting for part to part variations in leakage power ratios. This will bring the remaining variations to imperceptible levels.

The ratio of leakage power to total power is of particular interest in a modern high-performance microprocessor. Obviously this ratio is highly dependent on the application under test. Fig. 8 illustrates this ratio at Galgel's peak dissipation across a range of material.

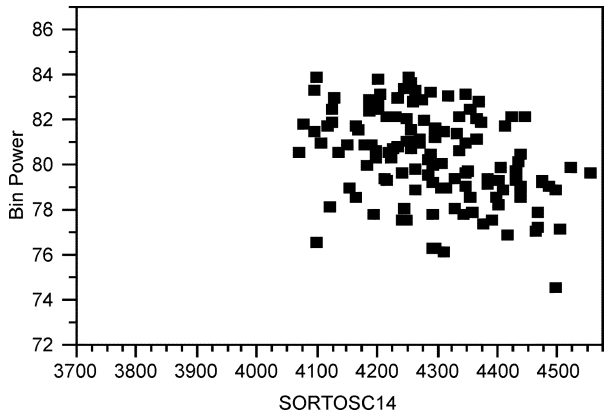


Fig. 6. De-rated power reference.

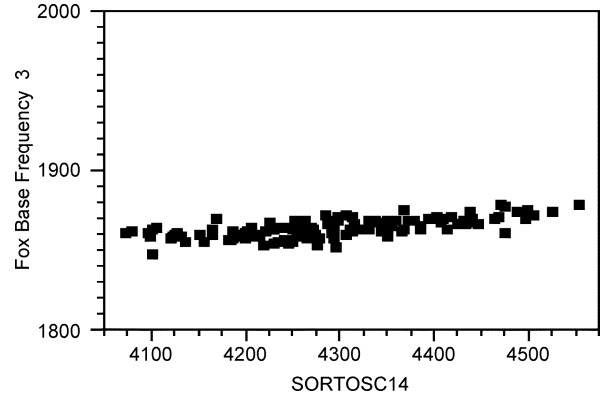


Fig. 7. Foxbase frequency.

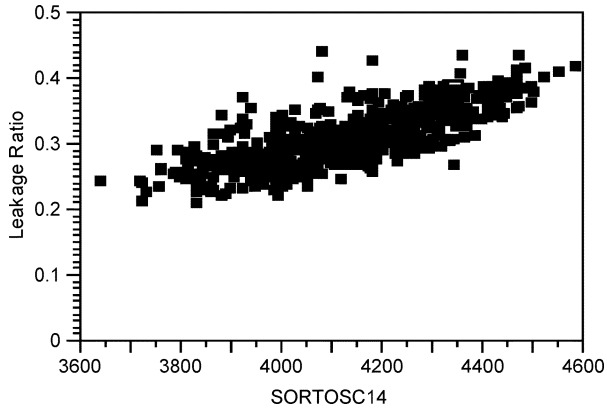


Fig. 8. Leakage power ratio.

III. Power-Supply Noise Measurement Circuits and Results

Another important aspect to power consumption and circuit robustness is the AC behavior of the on-chip supply voltage; to fully characterize the supply noise and its effects on circuits, both the noise distribution and its spectrum must be measured. Muhtaroglu [6] and Takamiya [7] have demonstrated subsampling circuits that can measure repetitive waveforms and distributions, and Alon [8] extended these techniques to measure the noise spectrum using only two samplers with simple VCO-based ADCs. Montecito included circuits based on Alon's work to quantify the supply noise and assist in the diagnosis of the dynamic frequency management system. A block diagram of the

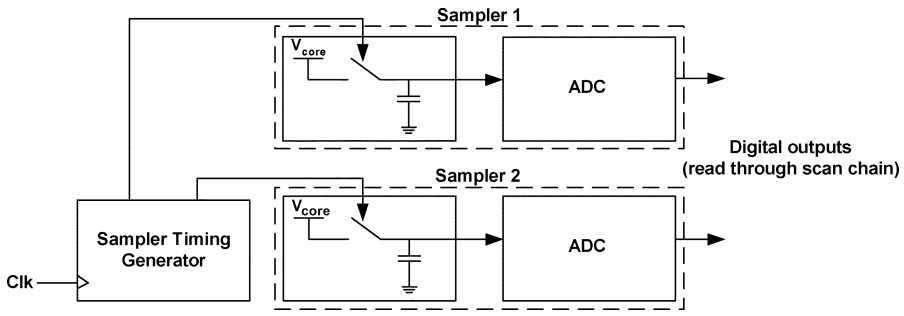


Fig. 9. Supply noise measurement system block diagram.

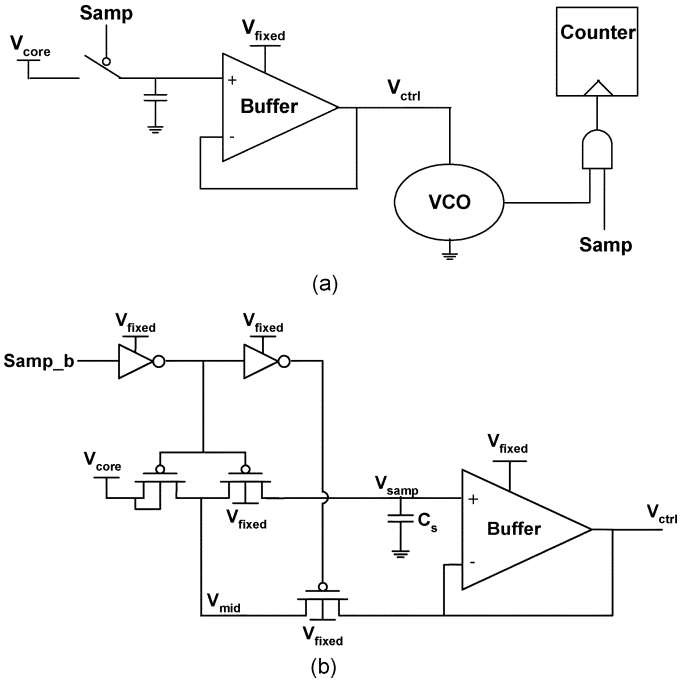


Fig. 10. (a) VCO based ADC block diagram; (b) sample and hold with source/drain leakage cancellation.

measurement system is shown in Fig. 9. In advanced technologies the sample and hold is one of the most challenging circuits to design, and therefore we first present circuits used to mitigate sampler leakage. We next describe the on-chip hardware used to generate the timing signals for the two samplers, and then present measured noise distributions and spectra.

A. Sample-and-Hold Design

As shown in Fig. 10(a), the VCO-based ADC uses the supply voltage sample to set the frequency of the VCO and then counts clock pulses over a fixed time window to estimate this frequency. The resolution of the ADC is set by the hertz-per-volt gain of the VCO and the width of the conversion window; to measure supply noise with a millivolt level resolution the window is usually on the order of hundreds of nanoseconds. Due to the unavailability of high threshold, thick oxide devices, the sampling switch had to be implemented using standard, high-leakage devices. To avoid undesirable filtering of the measurements, during hold mode the sampled voltage should be as independent of the current supply voltage as possible; with standard devices the leakage time constants were often shorter than the conversion window, making this isolation very difficult to achieve.

The sampler circuits were powered off of a separate, relatively quiet supply V_{fixed} that was in any case required for the clock generation circuitry. To mitigate the source/drain leakage of the sampling switch and hence improve its isolation, the sample and hold was modified from the previous design as shown in Fig. 10(b). The pMOS switch was split into two, and during hold mode the unity gain buffer forces V_{mid} to track V_{samp}. The voltage across the switch connected to C_s is hence nearly zero (limited by buffer offset), and the switch's source/drain leakage is eliminated.

Unlike source/drain leakage, the gate leakage of the sampling switch is of less concern because its effect on the sampled voltage is independent of future values of V_{core}, and hence it does not lead to filtering of the measurements. However, to maximize the ADCs resolution it is still desirable to keep the net leakage current low. While the gate leakage of the sampling switch could be eliminated in a manner similar to the source/drain leakage, this step was not taken because the gate leakage of the buffer opposes the gate leakage of the switch (V_{fixed} is raised above V_{core} when taking measurements to keep all of the devices in the buffer in saturation), and hence canceling the switch gate leakage often resulted in a lower leakage time constant.[1]

B. Sampler Timing Generation

To measure repetitive waveforms, time-dependent distributions (i.e. infinite persistence on an oscilloscope), or noise spectra, the sampler timing pulses need to be placed with relatively fine resolution within a time basis that spans the longest events of interest. As shown in Fig. 11, to meet these requirements we employed a coarse/fine architecture where the timing signals are placed with clock cycle granularity by a scan-chain controlled state machine (a counter and a set of comparators), and then adjusted with finer steps by a low-fanout inverter-based delay line. To avoid any dependency on the duty cycle of the clock, the state machine used only rising edges and the delay line was designed to cover at least one full clock cycle.

[1]The magnitudes of the two gate currents (from the sampling switch and from the buffer) are sensitive to sizing and process variations, and hence canceling the switch gate leakage can sometimes be beneficial. To cover both cases the cancellation needs to be programmable, but; the potential benefit in resolution was not worth this complexity.

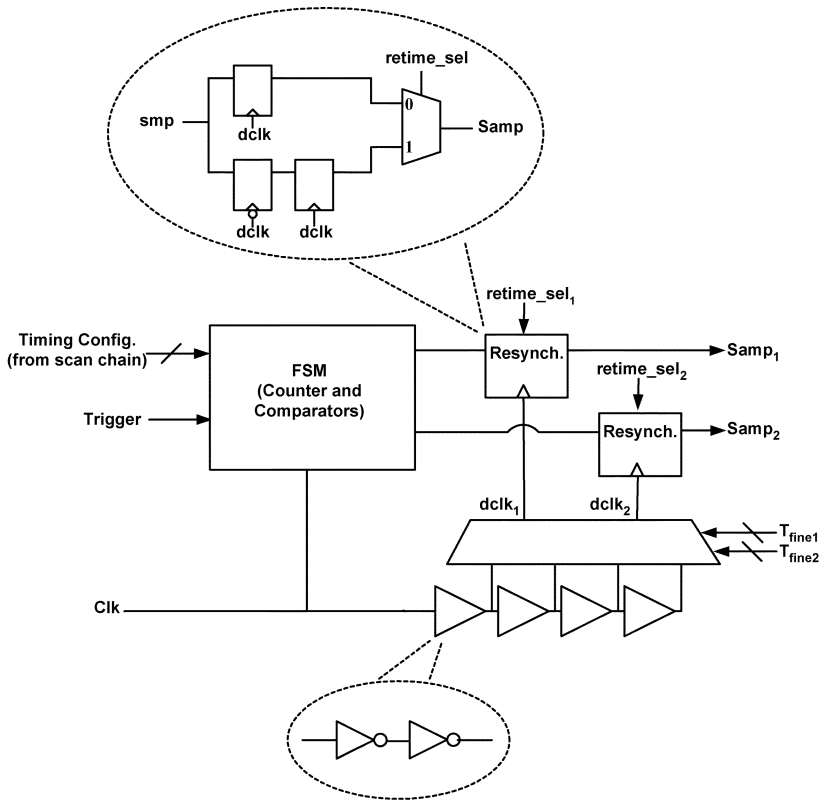


Fig. 11. Coarse/fine architecture for sampler timing generation.

Instead of using two separate delay lines for the two sampling signals, the clock was fed through a single delay line and the outputs of the state machine were resynchronized by the selected delayed clocks $(\mathrm{dclk}_{1,2})$ in order to minimize mismatches between the paths of the two sampling signals. For small delay settings, the smp signals from the state machine were retimed by the rising edge of dclk. As the delay of a dclk approaches a full cycle, the smp signal may not meet the setup or hold time of the rising-edge triggered resynchronization flip-flop—leading to indeterminism in the signal's timing. Therefore, for these delays smp is first captured by the falling edge of dclk before it is retimed by the rising edge. The setting where this occurs can be found through calibration with on-chip flip-flop phase detectors that determine the early/late relationship between Samp_1 and Samp_2; these detectors are in any case necessary to calibrate the delay line by measuring the number of delay stages that span a known clock cycle.

Since the fine delay is generated by a set of inverters that run off of V_{core}, the timing of the sampling signals will be affected by the noise that the circuits are measuring. While this does cause filtering of the true supply signal, as long as the delay sensitivity of the inverters is similar to that of the other circuits on the chip (mostly logic gates), the measurement circuits will see the noise with the same (supply-noise jittered) timing reference that the rest of the circuits do.

C. Measurement Results

Fig. 12 shows the distribution and spectrum obtained from measuring the noise with both cores operating in fixed frequency mode at 1.4 GHz with a nominal supply of 1.05 V; the cores were set to toggle between the power virus and low activity. As shown by the sharp dips in the distribution which repeat every clock cycle and the corresponding pulse train in the noise spectrum, a significant amount of noise is generated by clock-related activity that repeats every cycle. This is most likely because the flip-flops and many of the logic gates toggle at or near the rising edge of the clock, leading to a current profile with repetitive pulses whose magnitudes are modulated by the number and size of the transitioning gates.

The spectrum also shows an increase in the noise density at the lower frequencies ($<\sim$50 MHz) that is likely due to the resonance of the power distribution network. Since this lower frequency noise persists for many clock cycles, in a standard digital system the noise directly impacts performance because the critical paths must meet timing at the lowest voltage. One of the advantages of adaptive frequency control is that the chip adjusts itself to these variations and hence on average achieves higher performance. In this case, the measurements show $\sim$70 mV of peak-to-peak noise, indicating that along with its other advantages, dynamic frequency management can improve performance by $\sim$5% even in relatively quiet conditions [3].

IV. Clock-Insertion Delay Adjustment

After power-grid integrity, clocks are the next critical function to manage and optimize. We recognized early on that 90-nm variability and other factors would make clock-edge manipulation a critical capability. Therefore, the microprocessor implements a method to allow the clock insertion delay to the final level of clock buffering to be adjusted. This has proven to be a

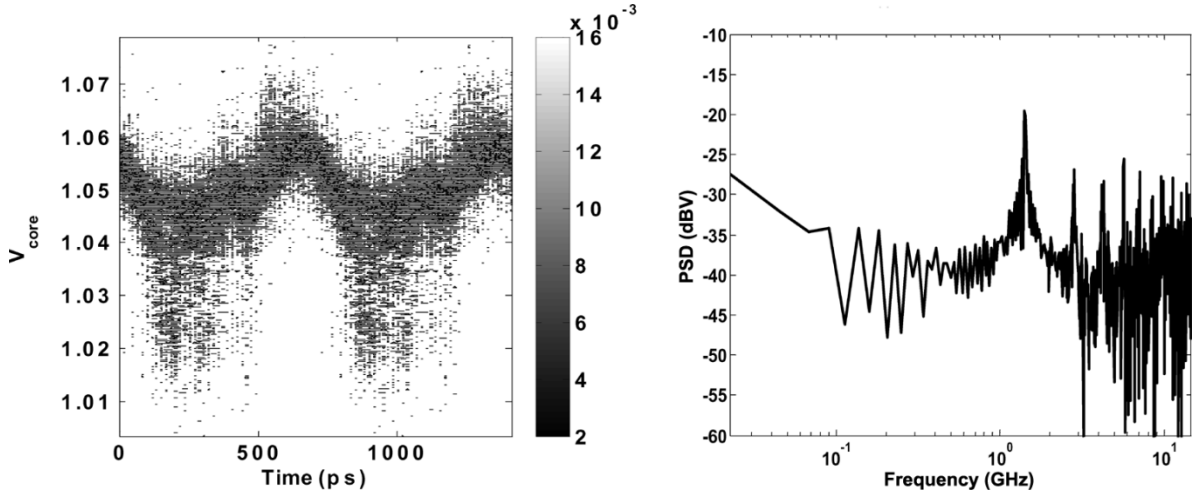


Fig. 12. Measured supply voltage distribution and spectrum with both cores toggling between high and low activity; the clock frequency is 1.4 GHz and Vcore is 1.05 V.

vital feature in supporting silicon debug and optimizing the frequency of the design.

In past implementations of Itanium microprocessors, one of the key learnings from silicon debug was that many silicon bug fixes tended to modify clock edges. By advancing or delaying clock edges, timing could be adjusted to fix a bug. Such fixes were often tested through FIB edits [4]. However, FIB edits can be unreliable, and are also difficult to execute in large volumes (which can sometimes be needed to enable progress if a silicon bug is blocking validation). The great benefit of being able to test out a silicon bug fix in volume prior to committing masks led to the idea of implementing a method to allow clock edges to be manipulated at will. This enables software checkout of possible silicon fixes, as well as a method of avoiding bugs in the current implementation. Such software modifications can even be used to permanently fix a bug, resulting in less effort required by the design team. Note that this capability was not used to compensate for any inaccuracy in pre-silicon clock skew analysis. The intent was to allow modification of clock edges to fix design errors that were not found with pre-silicon analysis tools.

In addition, it was expected that overall silicon frequency could be increased through "tuning," since the optimal clock setting for a local circuit is rarely one with zero skew compared to the rest of the processor. By using heuristic methods based upon static timing analysis to derive settings, and actual experimentation on silicon, an optimal setting for all clocks can be developed to optimize overall frequency as well.

A. CVD Implementation

A total of 6417 CVDs (Clock Vernier Devices) are present in each core, with another 388 in the logic shared between cores. A plot of the locations of each CVD for a particular core is shown in Fig. 13 to give an idea of the distribution throughout a core. Each CVD controls an average of 100 clocked elements.

A schematic of the CVD circuit is shown in Fig. 14. Each CVD receives a main clock signal (SLCBO) which is then delayed in a controlled manner (CVDO) and then delivered to the final buffering stage before latching circuit elements. Each CVDO controls one or more buffer stages. All clocks pass through a CVD before being utilized in the design; automatic checks were performed during design to guarantee this.

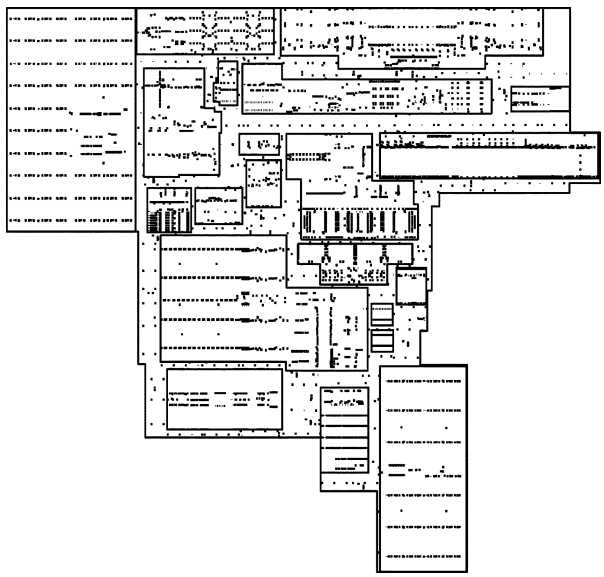


Fig. 13. CVD distribution in a core.

A controlled contention circuit implementation is currently being used, after discovery that the original implementation using simple switched FET capacitance had power consumption and granularity issues. The setting for each CVD is controlled by the three scanned control latches, allowing selection of eight different values of delay, which are shown in Table I. The CVDs are connected together into a serial scan chain. With all CVDs set to the maximum delay (not the default), the additional total power consumption due to the CVDs is approximately 1 W.

Adjustments can be performed by firmware override in a normal system, or directly via scan access for debugging purposes. When using firmware to override settings, during the initial power on reset the CVD scan chains are initialized to all zeros (minimal delay, which also results in minimal CVD power consumption). The processor executes code to retrieve the desired settings from off-chip and stores them in an internal

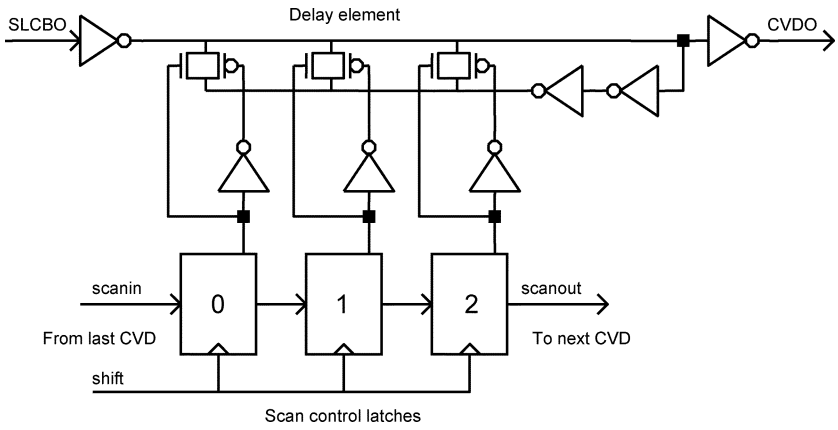


Fig. 14. Clock Vernier delay circuit implementation.

TABLE I
CVD DELAY (ps)

Delay Setting	000	001	010	011	100	101	110	111
Rising Delay	86.6	96.5	106.8	114.7	125.5	131.4	136.9	141.7
Falling Delay	86.1	95.6	106.4	114.2	126.6	132.3	138.0	142.7
Duty Cycle Error	0.6	0.8	0.4	0.5	1.0	-0.9	-1.1	-1.0
Step Distance	0.0	9.6	10.8	7.8	12.4	5.7	5.8	4.7

memory. An internal reset is then performed, and a state machine inside the processor takes control of the scan chain for the CVDs and shifts the values loaded into the internal memory via firmware into the scan chain. In the event that an all zeros setting does not allow enough chip functionality to program the internal memory, up to five individual CVDs can be addressed and programmed through fused settings to avoid the problem; this capability has not proven to be necessary.

For silicon debug purposes on a tester or in a system, the scan chain for the CVDs may also be overridden directly by scanning the desired values into the CVD scan chain. The processor can also be stalled through a special breakpoint mechanism at any point (even during operation of an OS), and the CVD values changed on the fly via scan, with execution resuming after the changes. This final capability has proven vital in debugging complex silicon bugs.

CVD settings are discovered through the process of speedpath and electrical debug. Once a speedpath or other problem is found, debug techniques such as scan dumps, clock manipulation and silicon probing are used to identify the failing circuit [5]. Once discovered, the CVDs that control the circuit's clocks can be changed to see if they fix the problem. Alternatively, CVDs can be used to find marginal circuits by changing the settings of CVDs algorithmically to see what effect individual CVDs have on tester or system content execution. Such experiments are conducted regularly during debug, and the best settings are accumulated into a weekly "recipe" that is tested extensively on the tester and in systems to determine if there are any adverse effects from or interactions between the individual CVD settings.

B. Results

CVDs have proven to be extremely beneficial. On initial Montecito silicon, CVDs were used to work around an electrical silicon bug which would have made it very difficult to boot an operating system at a reasonable operating point. CVDs proved to be key in helping to resolve the issue. A firmware "patch" enabled validation partners to avoid the bug entirely and proceed with functional and electrical validation, including booting three operating systems shortly after receiving first silicon.

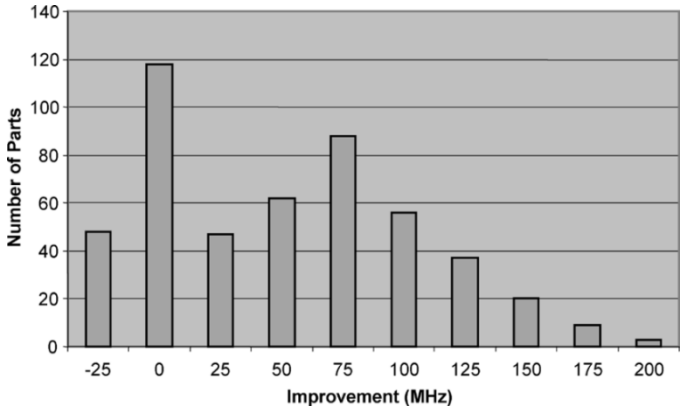


Fig. 15. Frequency improvement using CVDs.

This is a clear example of the benefit of the CVD capability. The design would not have functioned properly without the use of severely constraining code workarounds. It would have been necessary to quickly revise the design, which would have required a costly new mask set and could have set the program back by a few months. Alternatively, a FIB edit could have been performed to adjust the critical clocks, but again this could not have been executed in volume which would have limited validation. By using CVDs, the "fix" could be implemented on every part which had already been manufactured and validation could immediately proceed.

Another benefit has been in the improvement in frequency of operation as shown in Fig. 15. This is a histogram of the improvement in frequency seen when applying an early experimental CVD setting to 4% of the CVDs to a number of parts run through the manufacturing test flow. Note that a large number of the parts exhibit a significant improvement in frequency of operation. One interesting observation is that some parts experience no or negative benefit from the CVD settings. Debug has determined that this is due to CVD settings that are optimizing for

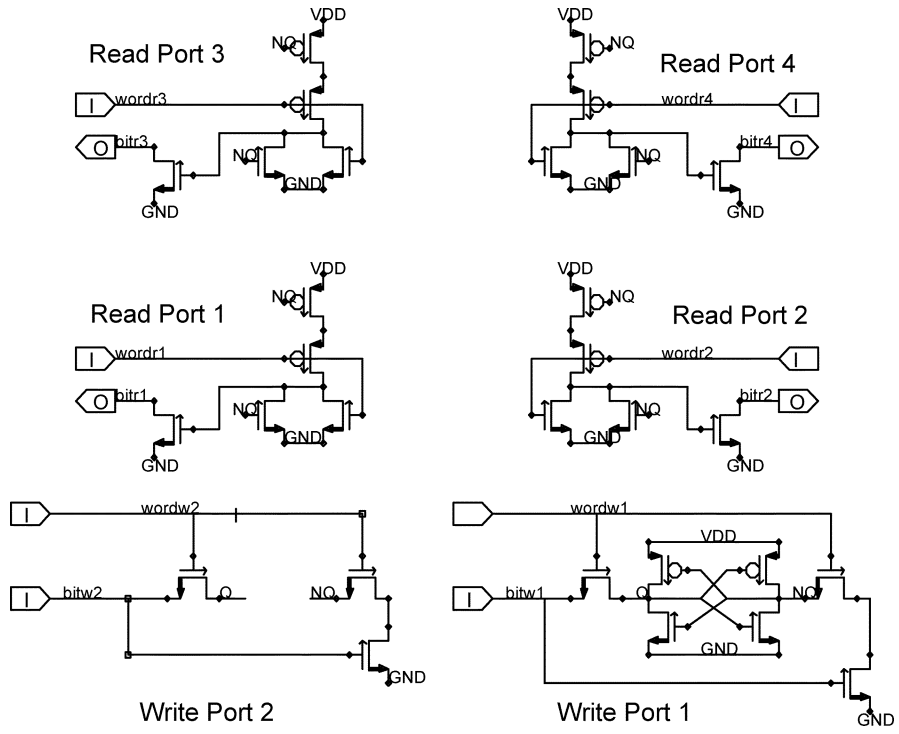


Fig. 16. Memory cell schematic.

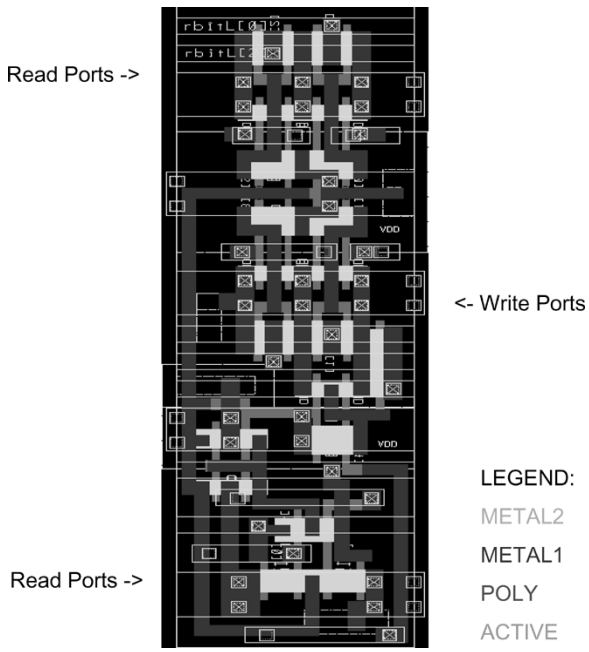


Fig. 17. Memory cell artwork.

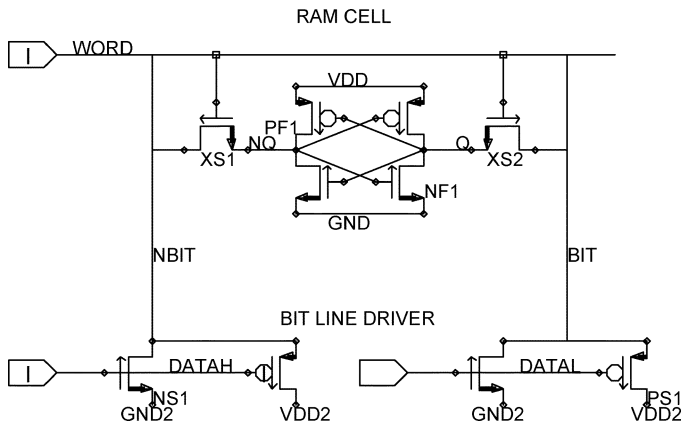


Fig. 18. Write robustness simulation schematic.

one path adversely affecting the previous or following path due to movement of the clock (i.e. a "back to back" speedpath). Even with the high degree of control provided by over 6000 CVDs, such cases still occur as some speedpaths are optimized across several phases or cycles. Even with this issue, CVDs have resulted in an average performance gain of over 100 MHz using more recent settings on most parts (approximately a 5% increase in frequency), which is very significant and expected to rise as tuning and experimentation methods improve.

V. Memory Array Methodology Improvements

Migrating circuits to 90 nm presented a number of challenges and the team focused on dealing with known troublesome areas. On the Itanium2 processor (180 and 130 nm), a multitude of full custom memory array circuits were used to design the three levels of cache hierarchy and periphery structures implemented in the architecture. These circuits were innovative, but not well regulated, and consequently were a common source of failure during silicon debug. Specifically, the design experienced a number of array write failures at low voltage and domino leakage issues at high temperatures and voltages. Due to the increased subthreshold leakage and transistor variability expected to be seen in the 90-nm process generation, a focused effort was made to improve the robustness and reliability of these cache circuits for Montecito.

A. Memory Array Library

For the Montecito microprocessor, a library was created that contained schematics of random access memory (RAM) cells and "bolt-on" interface circuits used for reading and write these cells. These library pieces were simulated with Spice and designed to be electrically robust across the entire process, voltage, and temperature space the microprocessor would see during silicon debug. Designers then were then enabled to assembled schematics of custom memory cell circuits using these library pieces and to built custom layouts for of these cells on user-defined wire pitches. Figs. 16 and 17 depicts the schematic and Fig. 17 depicts the layout of a 4 read port, 2 write port memory cell (2.64 u $\times$ 9.6 u) 2 read port and 2 write portused in a 32 entry by 144 bit queue that required a zero cycle load to use penalty. memory cell designed in this manner.

B. Robust Memory Array Writes

As mentioned previously, the Itanium2 processor had several low voltage memory array write-ability issues during post-silicon debug. This was not surprising, as determination of write stability for random access memory structures has long been an issue for integrated circuits. Traditionally, a statistical approach has been applied to guarantee write stability by applying a Monte Carlo SPICE simulation analysis to the transistor parameters that impact the ability of a memory cell to hold a value when written. Such a technique was utilized on the Itanium2 processor. Unfortunately, a statistical SPICE analysis of just the memory cells transistors has limitations. Though it guarantees that memory cells will hold their written values despite transistor variation assuming ideally driven bit lines, it does not take into account the circuitry used to write the memory cell. Voltage offsets between the memory cell and the write circuits, process variation of the bit line driver transistors, can both affect write robustness.

The Montecito processor implemented a new simulation method that expands on this previous work and ensures that the memory cell write is robust in the context of its surrounding circuitry. A simulation schematic is depicted in Fig. 18 for reference to illustrate this method. A stable six transistor memory cell is instantiated with its associated data bit line driver circuits. Different ground and supply voltages are connected to the write circuitry (VDD2 and GND2) to allow the modeling of a total 10% power grid offset from the memory cell power supply. Select transistors then have their drive currents skewed fast (or slow) in the direction to make the write difficult. For the case of writing a 1 to the "Q" node for the circuit, transistors PF1 and NF1 are skewed "fast" while the NS1, XS1, XS2, and PS1 are skewed "slow." Finally, Spice simulations are runs at the extremes of the post-silicon operation space to ensure the memory cell writes robustly at worst case low voltage conditions.

Post-silicon data indicates that this simulation technique has worked well. To date, no low voltage cache write issues have been found on Montecito silicon. A plot of voltage versus write frequency low voltage shmoo of a direct access cache test of the 12 MB L2D array is included in Fig. 19 to illustrate robust low-voltage operation of large cache arrays down to 0.7 V.

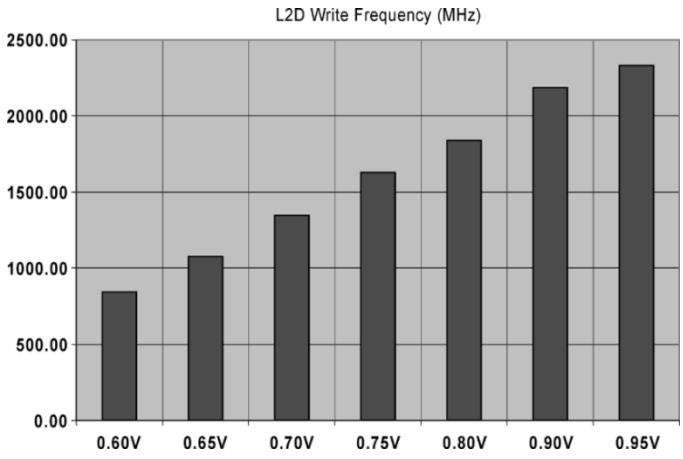


Fig. 19. Low-voltage memory array write shmoo.

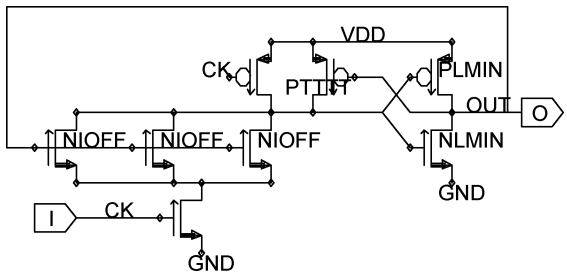


Fig. 20. Memory leakage simulation schematic.

C. Robust Burn-In Operation

During post-silicon validation, microprocessors are tested at elevated voltage and temperature during burn-in to find latent defects before parts are shipped to customers. Traditionally, large memory arrays on microprocessors with dynamic open-drain circuits are susceptible to low-frequency leakage issues in this operating space, and the Itanium2 processor was no exception. For the Montecito processor, correct functional operation at high voltage and low frequency during burn-in was required in order to run functional code to achieve its defect per million (DPM) manufacturing test goals.

The difficulty of making the Itanium2 memory array circuits leakage robust during burn-in was complicated by two issues. First, sub-threshold transistor leakage went up by 300% for nominal length transistors in the jump from a 130-nm process to a 90-nm process, requiring larger PFET transistor keepers that would not fit into the ported layout area. Second, sub-threshold leakage had become nonlinear with respect to transistor width, making the traditional pMOS keeper width to leaking nMOS width ratios inaccurate and overly pessimistic. The Montecito design team addressed these challenges in two ways. First, widespread use was made of nonminimum length channel nMOS devices (with 5$\times$ less sub-threshold leakage than minimum length devices) in memory arrays circuits; the transfer transistor on ram cells, pass transistors on column multiplexors, and stacked nMOS write transistors on domino

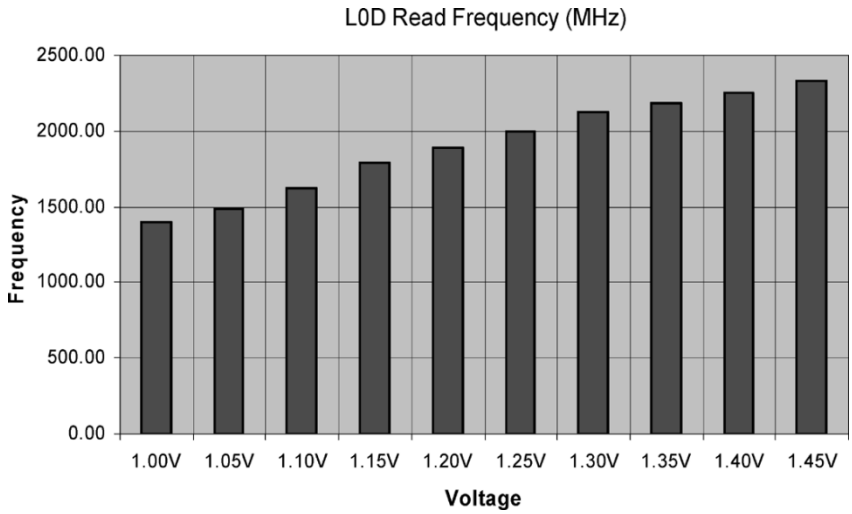


Fig. 21. High-voltage memory array read shmoo.

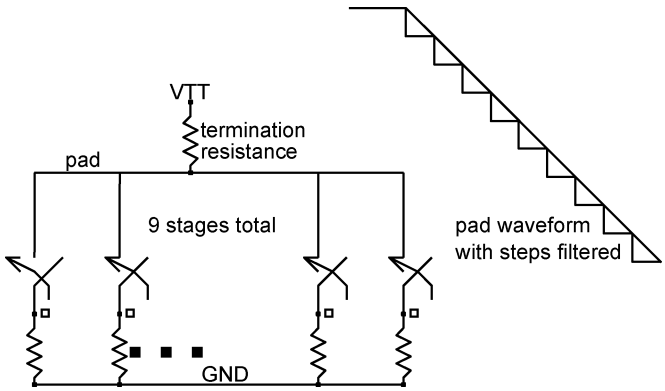


Fig. 22. Driver conceptual overview.

nodes were implemented with long channel devices to reduce leakage. Second, a Spice simulation pass-fail criterion was developed to guarantee uniform leakage robustness across the design that took into account the nonlinear behavior of leaking devices. Fig. 20 depicts a simulation schematic of this leakage pass/fail criteria in its simplest form. A domino circuit is simulated with nMOS pull-downs in the "ioff" (or leakiest) process skew, the output inverter in the "lmin" (or minimum length) process skew, and the pMOS keeper in the "tttt" (or typical) process skew. The domino gate self connects the output back to all inputs with the precharge node at a high value. The gate passes the leakage "flip test" if the pMOS keeper is large enough to prevent the domino note from leaking away at burn-in conditions.

For Montecito, this "flip test" was applied to all global dump circuits in the custom memory arrays and resulted in pfet keeper to leaking nfet width ratios of about 6%. Montecito silicon data has indicated this simulation technique has been very useful in making a good tradeoff between PFET keeper sizing and leakage robustness. The burn-in voltage versus frequency shmoo plot of the L0D memory array is depicted in Fig. 21 is illustrative of the fact that the Montecito processor is robust in this operating space.

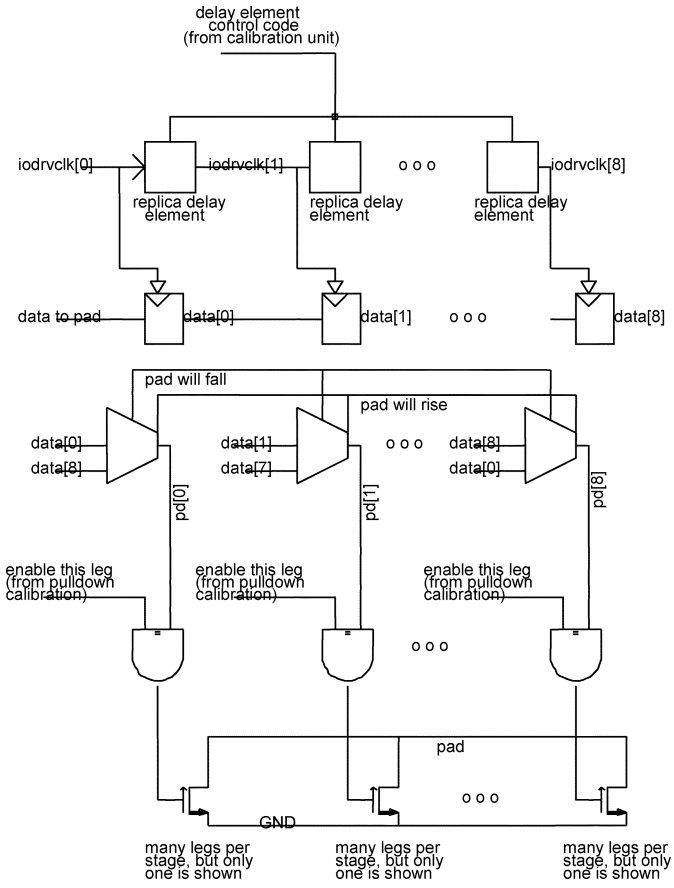


Fig. 23. Driver slew rate control.

VI. I/O Design

The final circuit challenge discussed here is that of enabling the legacy multi-drop bus technology to operate fast enough to feed the cores with sufficient memory bandwidth. Montecito has an AGTL+ front side bus interface, optimized for 45-Ω termination, which supports data rates from 400 MT/s to 800 MT/s

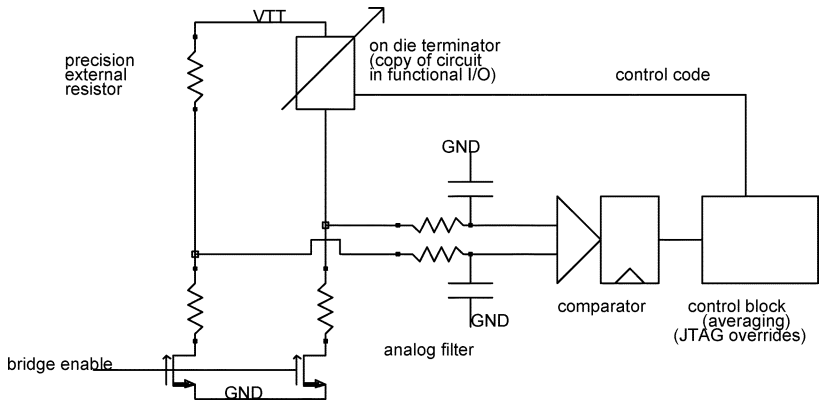


Fig. 24. On-die termination block.

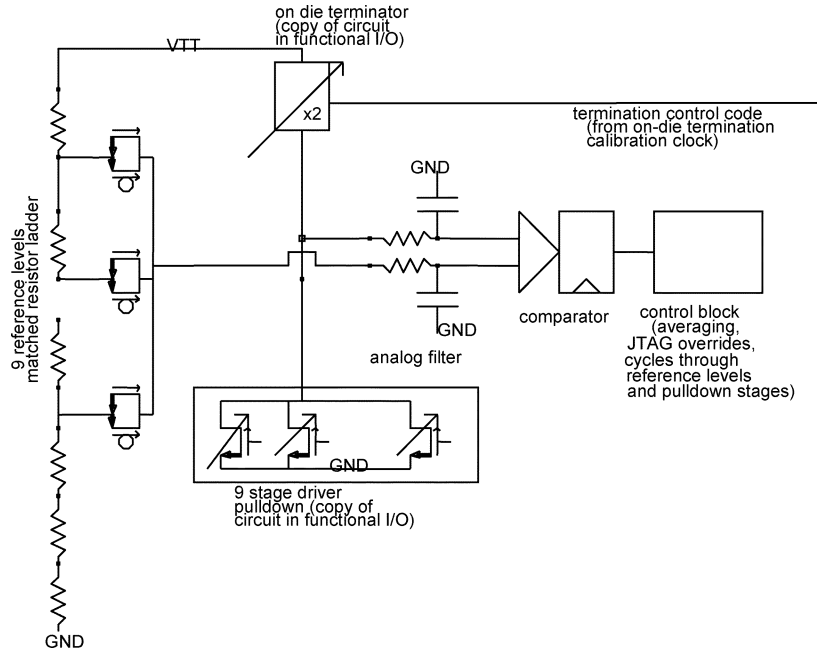


Fig. 25. Pulldown impedance calibration unit.

in 5-load, 3-load, and 2-load configurations. The I/Os are organized into 6 stripes as seen in the chip floor plan (Fig. 1). The three stripes on the right include the source synchronous data I/Os. The three stripes on the left include the control and address signals, which are globally synchronous signals which run at one half the rate of the data. The I/O circuits include circuits to calibrate the pull down impedance, on-die termination impedance, and the slew rate of the driver. Each of the six stripes includes its own independent calibration circuitry for I/O impedance and slew rate control. This section will review the driver operation and the calibration circuitry which supports it.

A. Driver Overview

Fig. 22 shows a conceptual overview of the driver. The driver's slew rate control is achieved by controlling the delay between turning on or off each of the nine consecutive stages of the driver's pull down. Each of the nine stages is calibrated so that when one is turned on, it will cause a 100 mV drop in the pad voltage when the pad is terminated with 22.5 Ω. This approach matches rising and falling delays and allows an optimum clock to out delay for a given slew rate, since the delay is dominated by the pad's transition time. Fig. 23 shows more detail of the driver circuit design and slew rate control. A single delay line is used to generate the pad clocks for about 10 I/Os. The driver includes a voltage converting mux to transition from the logic supply to the I/O supply and gates to turn on and off various pull down legs, as determined by the pull down impedance calibration loop. The mux is required because the stages must be turned off in the opposite order in which they are turned on so that the pad waveforms is linear. The number of extra gate delays in the clock to out delay are minimized and

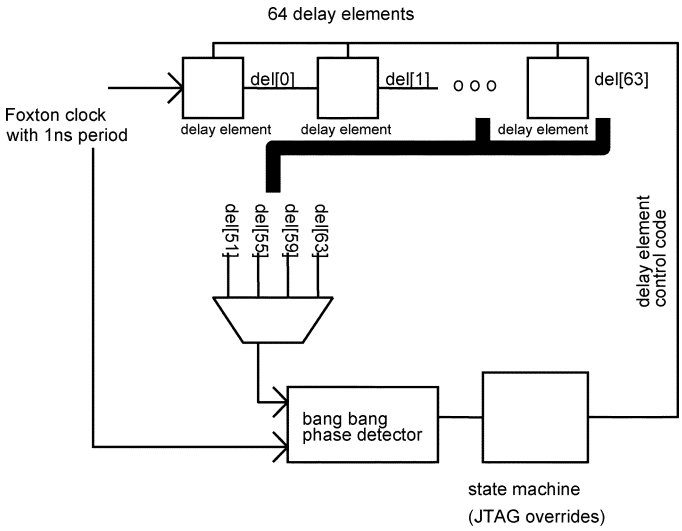


Fig. 26. Slew-rate delay calibration unit.

have fast edge rates; hence, they add minimal overhead to the clock to out delay for a given output slew rate.

B. Calibration Circuitry

Fig. 24 shows a schematic diagram for the on-die termination calibration unit. Each of the six stripes must share access to the precision off chip resistor. Each of the units will enable itself to pull down on the bridge for approximately 1 μs and then enable the next stripe in a round robin fashion. The bridge output is sampled at approximately 1 GHz after a 30-MHz *RC* filter. The control block then filters these samples and will update the thermometer coded termination bus.

The on-die termination circuit is composed of passive resistors gated by PFETs; the impedances are chosen to ensure linearity of less than better than 3% across the range of the pad's voltage swing; this linearity is critical in ensuring that the pad's slew rate is uniform across the voltage range.

Fig. 25 shows a schematic for the pull down impedance calibration unit. It is slaved from the on-die termination control because it pulls down against 2 copies of the on-die termination unit. Like the on-die termination unit, the control block samples at 1 GHz after an *RC* filter. These values are digitally filtered by the control block, which updates the driver strength for each stage and decides when to transition from one stage to the next. Since the driver pull down codes are not thermometer coded, these bits are only distributed to and latched at the I/O pads when they are tri-stated.

Fig. 26 shows a schematic for the slew rate delay calibration unit. It is a digitally controlled DLL which locks to the period of the fixed 1-GHz clock used by the Foxton unit. The base delay element is a current starved differential inverter. This digital DLL can select multiple taps to lock onto the 1 ns input period; the result is that we can select different delay element settings and hence different slew rates. The calibration unit then distributes the delay element control bits to replica delay element copies in the clock delay lines for the functional I/Os. Since these bits are not thermometer coded, the codes are only updated during safe periods in the bus clock cycle.

All calibration circuitry have full overrides available through JTAG for test and characterization of system margins as a function of slew rate and termination levels.

VII. Conclusion

The Montecito design team set out to achieve the goals of performance leadership in the high end server market combined with $> 3\times$ power efficiency increases and higher reliability for legacy system upgrades. These goals required significant innovation in power reduction, 90-nm circuit issues, clock control and I/O design. We have presented some of the innovations which were required to achieve those goals.

Acknowledgment

The authors recognize the extraordinary efforts of a committed design team in making the Montecito processor a success.

References

[1] S. Naffziger *et al.*, "The implementation of the Itanium2 microprocessor," *IEEE J. Solid-State Circuits*, vol. 37, no. 11, pp. 1448–1460, Nov. 2002.

[2] R. McGowen *et al.*, "Power and temperature control on a 90-nm Itanium family processor," *IEEE J. Solid-State Circuits*, vol. 41, no. 1, pp. 228–236, Jan. 2006.

[3] T. Fischer *et al.*, "A 90-nm variable frequency clock system for a power-managed Itanium architecture processor," *IEEE J. Solid-State Circuits*, vol. 41, no. 1, pp. 217–227, Jan. 2006.

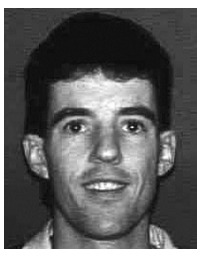
[4] R. H. Livengood and D. Medeiros, "Design for (physical) debug for silicon microsurgery and probing of flip-chip packaged integrated circuits," in *Proc. Int. Test Conf.*, 1999, pp. 880–882.

[5] D. D. Josephson, "The manic depression of microprocessor debug," in *Proc. Int. Test Conf.*, 2002, p. 659.

[6] A. Muhtaroglu, G. Taylor, and T. Rahal-Arabi, "On-die droop detector for analog sensing of power supply noise," *IEEE J. Solid-State Circuits*, vol. 39, no. 4, pp. 651–660, Apr. 2004.

[7] M. Takamiya, M. Mizuno, and K. Nakamura, "An on-chip 100-GHz sampling rate 8-channel sampling oscilloscope with embedded sampling clock generator," in *IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers*, vol. 1, Feb. 2002, pp. 182–458.

[8] E. Alon, V. Stojanović, and M. A. Horowitz, "Circuits and techniques for high-resolution measurement of on-chip power supply noise," *IEEE J. Solid-State Circuits*, vol. 40, no. 4, pp. 820–828, Apr. 2005.



Samuel Naffziger (M'02) received the B.S. degree in electrical engineering from the California Institute of Technology, Pasadena, in 1988, and the M.S.E.E. degree from Stanford University, Stanford, CA, in 1993.

He joined Hewlett Packard in 1988, and spent eight years working on various aspects of the PA-RISC processor development including floating point out-of-order execution and circuit methodologies. He then became part of the Itanium2 Joint Development Team with Intel Corporation, Fort Collins, CO, and has led the design of both the first Itanium2 processor, and most recently, the Montecito design. He is currently the Director of Itanium Circuits and Technology within Intel. He holds 59 U.S. patents on processor circuits and architecture.

Mr. Naffziger chairs the International Solid-State Circuits Conference Digital Subcommittee, and is an Intel Fellow.

Blaine Stackhouse (M'04) received the B.S.E.E. degree from Virginia Polytechnic Institute and State University, Blacksburg, in 1987, and the M.S.E.E. from North Carolina State University, Raleigh, NC, in 1989.

He worked for the Digital Equipment Corporation and designed CPU peripheral chips including the first PCI–PCI bridge. He joined Hewlett Packard in 1994 and has worked on Itanium processors since 1996 as a Unit Lead on both the McKinley and Montecito processors. He is currently a Principal Engineer at Intel Corporation, Fort Collins, CO, leading the implementation of the next generation Itanium processor. He holds three U.S. patents with several pending, and has authored several conference papers.

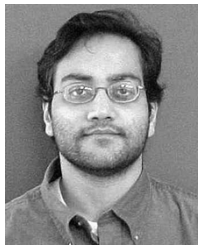
Tom Grutkowski received the B.S. degree from The Cooper Union for the Advancement of Science and Art, New York, in 1987, and the Master's degree from the Georgia Institute of Technology, Atlanta, in 1992.

He has been with Intel Corporation, Fort Collins, CO, since 1994, contributing to the Pentium II, Itanium, and Itanium2 design efforts. In addition to the microprocessor design effort, his interests extend to post-silicon debug and characterization. He is currently working on the next-generation IPF processor. He holds patents for both arithmetic and cache designs.

Doug Josephson received the B.S.E.E. degree from the University of Iowa, Ames, in 1988.

He joined Hewlett-Packard (HP) in 1989, where he was responsible for design, testing and debug of 5 PA-RISC processors. In 1996, he joined the Itanium development effort between HP and Intel Corporation, leading the test and debug methodology and silicon debug of the first Itanium2 processor as well as the Montecito design. He is currently a Senior Principal Engineer at Intel Corporation, Fort Collins, CO. He has authored a number of journal and conference papers. He is a coauthor of an upcoming book chapter on silicon debug to be published in 2006. He holds nine U.S. patents with several pending.

Mr. Josephson received the honorable mention paper award at the International Test Conference in 2001.



Jayen Desai received the B.S. and M.S. degrees in electrical engineering from Stanford University, Stanford, CA, in 1994.

From 1995 to 2005, he worked at Hewlett Packard, Fort Collins, CO, on several PA-RISC and Itanium2 processors in the areas of timing analysis, clock design, synchronizer design, custom digital design, and I/O design. He is now a Staff Engineer at the Design Center, Intel Corporation, Fort Collins, CO, and is working on future Itanium2 processors.

Elad Alon (SM'02) was born in Haifa, Israel, in 1979. He received the B.S. and M.S. degrees in electrical engineering from Stanford University, Stanford, CA, in 2001 and 2002, respectively, and is currently working toward the Ph.D. degree at Stanford University.

He was a Visiting Researcher at Hewlett Packard, Fort Collins, CO, in the summer of 2003, where he implemented supply noise measurement circuits on an Itanium processor. He has also been a Visiting Researcher at Rambus, Inc., Los Altos, CA, since 2003, where he has worked on supply noise measurement and high-speed signaling circuits. His research interests include efficient on-chip power supply regulation and distribution, noise measurement techniques, high-speed interface design, and applications of optimization and nonlinear control to high-speed mixed-signal circuits.

Mark Horowitz (S'77–M'78–SM'95–F'00) received the B.S. and M.S. degrees in electrical engineering from the Massachusetts Institute of Technology, Cambridge, in 1978, and the Ph.D. degree from Stanford University, Stanford, CA, in 1984.

He is the Yahoo Founder's Professor of Electrical Engineering and Computer Science at Stanford University. He has also worked in a number of other chip design areas including high-speed and low-power memory design, high-bandwidth interfaces, and fast floating point. In 1990 he took leave from Stanford to help start Rambus Inc, a company designing high-bandwidth memory interface technology. His research area is in digital system design, and he has led a number of processor designs including MIPS-X, one of the first processors to include an on-chip instruction cache, TORCH, a statically scheduled superscalar processor that supported speculative execution, and FLASH, a flexible DSM machine. His current research includes multiprocessor design, low-power circuits, memory design, and high-speed links.

Dr. Horowitz received the Presidential Young Investigator Award and an IBM Faculty development award in 1985. In 1993, he received the Best Paper Award at the IEEE International Solid-State Circuits Conference.